# *Introduction to Computer Algebra*

## *lecture II*

**Sven-Olaf Moch**

*Universität Hamburg*

– Computer Algebra and Particle Physics 2015, Hamburg, March 23, 2015 –

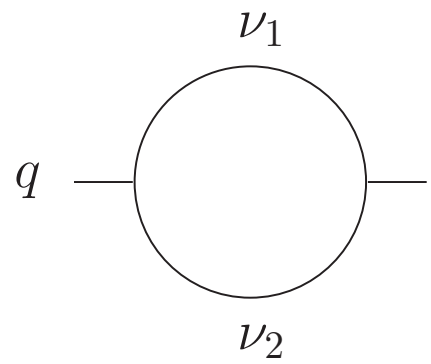
# *Plan*

- Particle physics and perturbation theory at work
  *Monday, March 23, 2015*
- *Symbolic sums and recurrence relations*
  *Tuesday, March 24, 2015*
- Polylogarithms and generating functions
  *Thursday, March 24, 2015*

# *Loop integrals*

- Easy example:
  massless one-loop two-point function $\mathrm{L1}$

$$\mathrm{L1} \;=\; \int d^D p_1 \, \frac{1}{(p_1^2)^{\nu_1} \, ((p_1 - q)^2)^{\nu_2}}$$



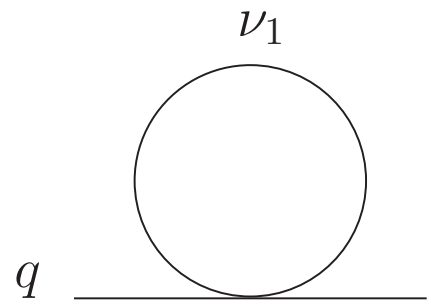

- Results for $\mathrm{L1}$

$$\begin{aligned} \mathrm{L1} \quad &= \quad \mathrm{i}\,(-1)^{\nu_1+\nu_2}\,\pi^{-D/2}\,(-p^2)^{D/2-\nu_1-\nu_2} \;\times \\ &\quad \times \frac{\Gamma(\nu_1+\nu_2-D/2)\Gamma(D/2-\nu_1)\Gamma(D/2-\nu_2)}{\Gamma(\nu_1)\Gamma(\nu_2)\Gamma(D-\nu_1-\nu_2)} \end{aligned}$$

- Expansion of Gamma-function in $\epsilon = 2 - \frac{D}{2}$ around positive integers values ($\nu_i \geq 0$)
  - Riemann zeta values $\Gamma(1+\epsilon) = 1 - \epsilon\gamma_E + \dfrac{\epsilon^2}{2}(\zeta_2 + \gamma_E^2) + \ldots$
  - $\overline{\mathrm{MS}}$-scheme puts $\gamma_E = 0$

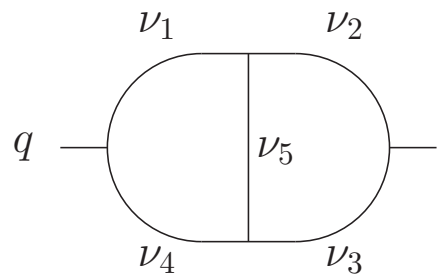
- Easiest example: massless tadpole (scaleless integral)
  - vanishes in dimensional regularization

$$0 = \int d^D p_1 \, \frac{1}{(p_1^2)^{\nu_1}}$$



- Example: massless two-loop two-point function

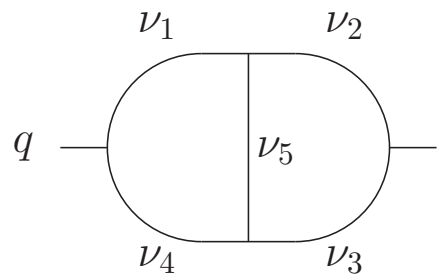
$$\mathrm{T1} = \int d^D p_1 \, d^D p_2 \, \frac{1}{(p_1^2)^{\nu_1} \, (p_2^2)^{\nu_2} \, (p_3^2)^{\nu_3} \, (p_4^2)^{\nu_4} \, (p_5^2)^{\nu_5}}$$




- $S_6 \times Z_2$ symmetry group of $\mathrm{T1}$ (1440 elements)

- Example: massless two-loop two-point function

$$\mathrm{T1} = \int d^D p_1\, d^D p_2\, \frac{1}{(p_1^2)^{\nu_1}\,(p_2^2)^{\nu_2}\,(p_3^2)^{\nu_3}\,(p_4^2)^{\nu_4}\,(p_5^2)^{\nu_5}}$$



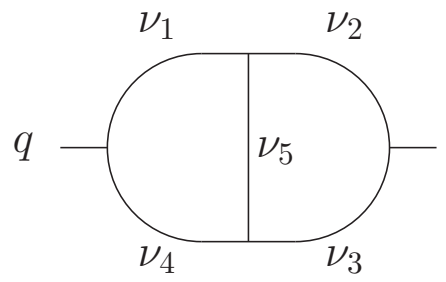

- $S_6 \times Z_2$ symmetry group of $\mathrm{T1}$ (1440 elements)

## *History*

- Long time it was not clear whether multiple zeta values are sufficient for coefficients in Taylor expansion in $\epsilon = 2 - \frac{D}{2}$

- Example: massless two-loop two-point function

$$\mathrm{T1} = \int d^D p_1\, d^D p_2\, \frac{1}{(p_1^2)^{\nu_1}\,(p_2^2)^{\nu_2}\,(p_3^2)^{\nu_3}\,(p_4^2)^{\nu_4}\,(p_5^2)^{\nu_5}}$$



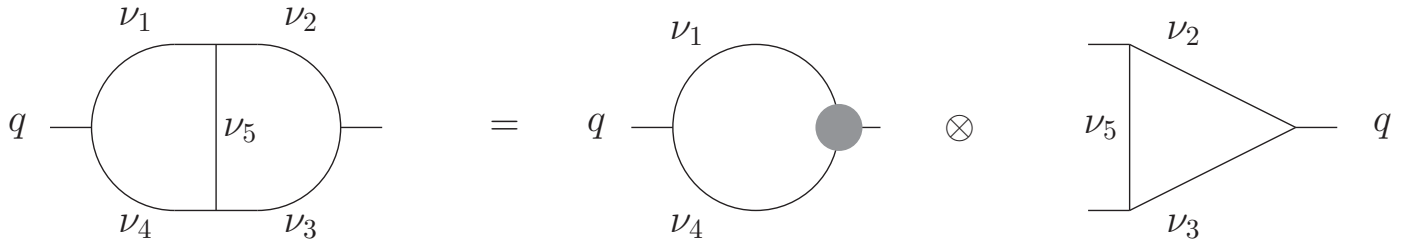

- $S_6 \times Z_2$ symmetry group of $\mathrm{T1}$ (1440 elements)

## *History*

- Long time it was not clear whether multiple zeta values are sufficient for coefficients in Taylor expansion in $\epsilon = 2 - \frac{D}{2}$
- History (all powers of propagators $\nu_i = 1 + \epsilon$)
  - $\epsilon^3$, $\epsilon^4$ $(\zeta_6, \zeta_7)$ Kazakov '84,Kazakov '85
  - $\epsilon^5$ $(\zeta_8, \zeta_{6,2})$ Broadhurst '86
  - $\epsilon^6$ $(\zeta_9)$ Barfoot, Broadhurst '88
  - $\epsilon^7$,$\epsilon^8$ $(\zeta_{10}, \zeta_{8,2}, \zeta_{11}, \zeta_{8,2,1})$ Broadhurst, Gracey, Kreimer '96
  - $\epsilon^9$ $(\zeta_{12}, \zeta_{10,2}, \zeta_{8,2,1,1})$ Broadhurst '02
  - $\epsilon^{10}$,$\epsilon^{11}$, $\epsilon^{12}$,$\epsilon^{13}$ Vermaseren '03 [based on Bierenbaum, Weinzierl '03]

# *Solution*

- Multiple zeta values are sufficient for the Laurent expansion of the two-loop integral $\mathrm{T1}(D, \nu_1, \nu_2, \nu_3, \nu_4, \nu_5)$, if all powers of the propagators are of the form $\nu_j = n_j + a_j \epsilon$, where the $n_j$ are positive integers and the $a_j$ are non-negative real numbers.
  Bierenbaum, Weinzierl '03
  - Convolution by means of Mellin-Barnes representation



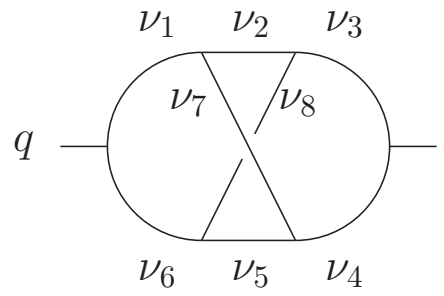

- Mellin-Barnes integrals by sum over residues

$$\mathrm{T1}(D, \nu_1, \nu_2, \nu_3, \nu_4, \nu_5) =$$

$$\frac{1}{(2\pi i)^2} \int\limits_{\gamma_1 - i\infty}^{\gamma_1 + i\infty} d\sigma \int\limits_{\gamma_2 - i\infty}^{\gamma_2 + i\infty} d\tau \, \mathrm{L1}(D, \nu_1 - \tau, \nu_4 - \sigma) \, \mathrm{MB}\left[\mathrm{Tri}_1(D, \nu_2, \nu_3, \nu_5; \tau, \sigma)\right]$$

# *Example (a bit more complicated)*

- Massless non-planar three-loop two-point function $\mathrm{NO}$




- "Master integral"
  (all indices $\nu_i = 1$, $D = 4 - 2\epsilon$)

$$\mathrm{NO} =$$

Chetyrkin, Kataev, Tkachev '80 $\quad +20\zeta_5$

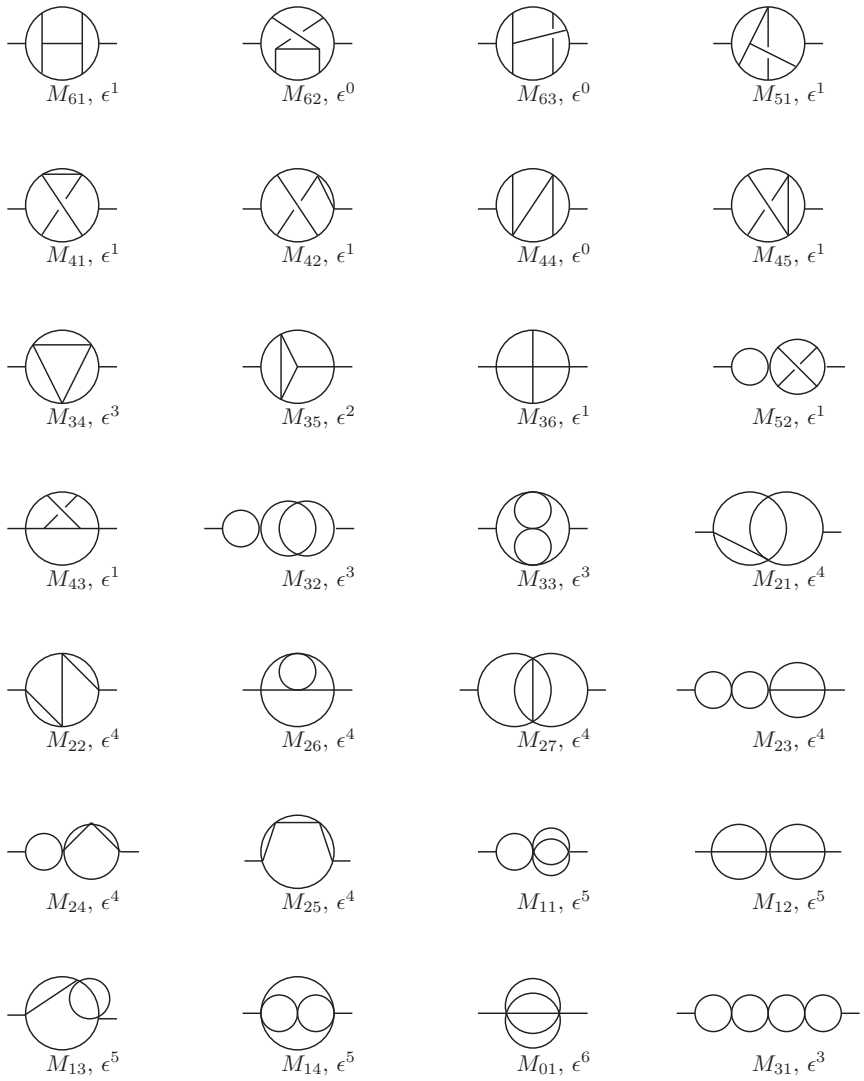
Kazakov '85 $\quad +\epsilon\left(68\zeta_3^2 - 80\zeta_5 + 50\zeta_6\right)$

Chetyrkin '04; Bekavac '05 $\quad +\epsilon^2\left(-272\zeta_3^2 + 204\zeta_3\zeta_4 + 80\zeta_5 - 200\zeta_6 + 450\zeta_7\right)$

- Technology used includes (no analysis from first principles yet)
  - Gegenbauer polynomials in $x$-space
  - Uniqueness relations
  - Cutting-and-glueing

# *Solution up to four loops*

- All massless propagators up to four loops are linearly reducible. Their $\epsilon$-expansions only contain alternating Euler sums $\mathrm{Li}_{n_1,\ldots,n_r}(\pm 1,\ldots,\pm 1)$
  Brown '09; Panzer '13



# *Computer algebra*

## *Goal*

- Computer algebra system which takes any Feynman integral as input and produces answer subject to following criteria:
  - completeness: every aspect of mathematics is covered
  - intelligence: output of all solutions we ever wanted to have
  - efficiency in performance: solutions on the spot
  - short development cycles: interactive use and high-quality output

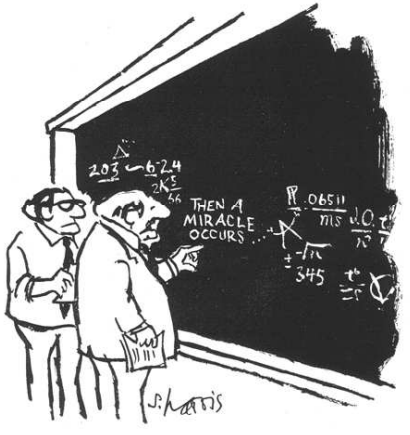
# Computer algebra

## Goal

- Computer algebra system which takes any Feynman integral as input and produces answer subject to following criteria:
  - completeness: every aspect of mathematics is covered
  - intelligence: output of all solutions we ever wanted to have
  - efficiency in performance: solutions on the spot
  - short development cycles: interactive use and high-quality output

## Challenge

- Unfortunately, life is not easy




"I THINK YOU SHOULD BE MORE EXPLICIT HERE IN STEP TWO."

# *Symbolic Summation*

## *Polynomial summation*

- Examples

$$\sum_{i=0}^{n-1} i = \frac{1}{2}n(n-1)$$

$$\sum_{i=0}^{n-1} i^2 = \frac{1}{6}n(n-1)(2n-1)$$

$$\sum_{i=0}^{n-1} i^3 = \frac{1}{4}n^2(n-1)^2$$

$$\sum_{i=0}^{n-1} i^4 = \frac{1}{30}n(n-1)(2n-1)(3n^2-3n-1)$$

# *Difference operator*

- Introduce operator $\Delta$ with $(\Delta f)(n) = f(n+1) - f(n)$
- If $g = (\Delta f)$, then (for $a, b \in \mathbf{N}, a \leq b$)

$$\sum_{i=a}^{b-1} g(i) \qquad = \sum_{i=a}^{b-1} \big(f(i+1) - f(i)\big)$$

# *Difference operator*

- Introduce operator $\Delta$ with $(\Delta f)(n) = f(n+1) - f(n)$
- If $g = (\Delta f)$, then (for $a, b \in \mathbf{N}, a \leq b$)

$$\sum_{i=a}^{b-1} g(i) \qquad = \sum_{i=a}^{b-1} \big(f(i+1) - f(i)\big) = \sum_{i=a}^{b-1} f(i+1) - \sum_{i=a}^{b-1} f(i)$$

# *Difference operator*

- Introduce operator $\Delta$ with $(\Delta f)(n) = f(n+1) - f(n)$
- If $g = (\Delta f)$, then (for $a, b \in \mathbf{N}, a \leq b$)

$$\begin{aligned}\sum_{i=a}^{b-1} g(i) &= \sum_{i=a}^{b-1} \big(f(i+1) - f(i)\big) = \sum_{i=a}^{b-1} f(i+1) - \sum_{i=a}^{b-1} f(i) \\ &= \sum_{i=a+1}^{b} f(i) - \sum_{i=a}^{b-1} f(i)\end{aligned}$$

# *Difference operator*

- Introduce operator $\Delta$ with $(\Delta f)(n) = f(n+1) - f(n)$
- If $g = (\Delta f)$, then (for $a, b \in \mathbf{N}, a \leq b$)

$$\begin{aligned}\sum_{i=a}^{b-1} g(i) &= \sum_{i=a}^{b-1} \big(f(i+1) - f(i)\big) = \sum_{i=a}^{b-1} f(i+1) - \sum_{i=a}^{b-1} f(i) \\ &= \sum_{i=a+1}^{b} f(i) - \sum_{i=a}^{b-1} f(i) = f(b) - f(a)\end{aligned}$$

# *Difference operator*

- Introduce operator $\Delta$ with $(\Delta f)(n) = f(n+1) - f(n)$
- If $g = (\Delta f)$, then (for $a, b \in \mathbf{N}, a \leq b$)

$$\begin{aligned}\sum_{i=a}^{b-1} g(i) &= \sum_{i=a}^{b-1} \big(f(i+1) - f(i)\big) = \sum_{i=a}^{b-1} f(i+1) - \sum_{i=a}^{b-1} f(i) \\ &= \sum_{i=a+1}^{b} f(i) - \sum_{i=a}^{b-1} f(i) = f(b) - f(a)\end{aligned}$$

- Consecutive cancellation of summands: telescoping
- Symbolic summation problem
  $g = (\Delta f)$ with $f = (\sum g)$, operator $\Delta$ is left inverse $\Delta(\sum f) = f$
- Cf. symbolic integration (differential operator $D$)

$$g = Df = \frac{d}{dx} f \quad \longrightarrow \quad \int_a^b dx g(x) = f(b) - f(a)$$

# *Difference operator (cont'd)*

- Differential operator $D$ acts in continuum as $D(x^m) = mx^{m-1}$

# *Difference operator (cont'd)*

- Differential operator $D$ acts in continuum as $D(x^m) = mx^{m-1}$
- Action of discrete analog $\Delta$ on polynomials?

# *Difference operator (cont'd)*

- Differential operator $D$ acts in continuum as $D(x^m) = mx^{m-1}$
- Action of discrete analog $\Delta$ on polynomials?
  - Example: $\Delta(n^3) = 3n^2 + 3n - 1$

## *Difference operator (cont'd)*

- Differential operator $D$ acts in continuum as $D(x^m) = mx^{m-1}$
- Action of discrete analog $\Delta$ on polynomials?
  - Example: $\Delta(n^3) = 3n^2 + 3n - 1$

## *Rising and falling factorials*

- Define rising factorials as $f^{\overline{m}} = f(x)f(x+1)\ldots f(x+m-1)$ (also known as Pochhammer symbols $(x)_m$ )

## *Difference operator (cont'd)*

- Differential operator $D$ acts in continuum as $D(x^m) = mx^{m-1}$
- Action of discrete analog $\Delta$ on polynomials?
  - Example: $\Delta(n^3) = 3n^2 + 3n - 1$

## *Rising and falling factorials*

- Define falling factorials as $f^{\underline{m}} = f(x)f(x-1)\ldots f(x-m+1)$

## *Difference operator (cont'd)*

- Differential operator $D$ acts in continuum as $D(x^m) = mx^{m-1}$
- Action of discrete analog $\Delta$ on polynomials?
  - Example: $\Delta(n^3) = 3n^2 + 3n - 1$

## *Rising and falling factorials*

- Define falling factorials as $f^{\underline{m}} = f(x)f(x-1)\ldots f(x-m+1)$
- Then, with falling factorials

$$\Delta(x^{\underline{m}}) = mx^{\underline{m-1}} \qquad \sum_{i=0}^{n-1} i^{\underline{m}} = \frac{1}{m+1} n^{\underline{m+1}}$$

- Conversion of polynomial powers $x^m$
  (decomposition with Stirling numbers of second kind $\left\{ \begin{array}{c} m \\ i \end{array} \right\}$ )

$$x^m = \sum_{i=0}^{m} \left\{ \begin{array}{c} m \\ i \end{array} \right\} x^{\underline{i}}$$

  - Stirling numbers of second kind denote # of ways to partition $n$ things in $k$ non-empty sets

# *Examples*

- Polynomials

$$\sum_{i=0}^{n-1} i = \sum_{i=0}^{n-1} i^{\underline{1}} = \frac{1}{2} n^{\underline{2}} = \frac{1}{2} n(n-1)$$

$$\sum_{i=0}^{n-1} i^2 = \sum_{i=0}^{n-1} (i^{\underline{2}} + i^{\underline{1}}) = \frac{1}{3} n^{\underline{3}} + \frac{1}{2} n^{\underline{2}} = \frac{1}{6} n(n+1)(2n+1)$$

$$\sum_{i=0}^{n-1} i^3 = \sum_{i=0}^{n-1} (i^{\underline{3}} + 3i^{\underline{2}} + i^{\underline{1}}) = \frac{1}{4} n^{\underline{4}} + n^{\underline{3}} + \frac{1}{2} n^{\underline{2}} = \frac{1}{4} n^2 (n+1)^2$$

# *Hypergeometric summation*

## *Definition*

- Hypergeometric function ${}_mF_n$

$$ {}_mF_n\left(\begin{array}{c} a_1,\ldots,a_m \\ b_1,\ldots,b_n \end{array}\middle| z\right) = \sum_{i\geq 0} \frac{a_1^{\overline{i}}\ldots a_m^{\overline{i}}}{b_1^{\overline{i}}\ldots b_n^{\overline{i}}}\,\frac{z^i}{i!} $$

# Hypergeometric summation

## Definition

- Hypergeometric function ${}_mF_n$

$$ {}_mF_n\left(\begin{array}{c} a_1,\ldots,a_m \\ b_1,\ldots,b_n \end{array}\middle| z\right) = \sum_{i\geq 0} \frac{a_1^{\overline{i}}\ldots a_m^{\overline{i}}}{b_1^{\overline{i}}\ldots b_n^{\overline{i}}}\,\frac{z^i}{i!} $$

## Examples

$$ {}_0F_0\left(\;\middle| z\right) = \sum_{i\geq 0}\frac{z^i}{i!} = \exp(z) $$

$$ {}_2F_1\left(\begin{array}{c} a,1 \\ 1 \end{array}\middle| z\right) = \sum_{i\geq 0} a^{\overline{i}}\,\frac{z^i}{i!} = \frac{1}{(1-z)^a} $$

$$ {}_2F_1\left(\begin{array}{c} 1,1 \\ 2 \end{array}\middle| z\right) = z\sum_{i\geq 0}\frac{1^{\overline{i}}1^{\overline{i}}}{2^{\overline{i}}}\,\frac{z^i}{i!} = -\ln(1-z) $$

# *Ratios*

- A term $g_n$ is hypergeometric, if the ratio $r(n)$ of two consecutive terms is a rational function of $n$.

$$r(n) = \frac{g_{n+1}}{g_n}$$

# *Ratios*

- A term $g_n$ is hypergeometric, if the ratio $r(n)$ of two consecutive terms is a rational function of $n$.

$$r(n) = \frac{g_{n+1}}{g_n}$$

- Example: binomial coefficient

$$\frac{\binom{m}{n+1}}{\binom{m}{n}} = \frac{\Gamma(m+1)\Gamma(n+1)\Gamma(m-n+1)}{\Gamma(n+2)\Gamma(m-n)\Gamma(m)} = \frac{-n+m}{n+1}$$

# *Ratios*

- A term $g_n$ is hypergeometric, if the ratio $r(n)$ of two consecutive terms is a rational function of $n$.

$$r(n) = \frac{g_{n+1}}{g_n}$$

- Example: binomial coefficient

$$\frac{\binom{m}{n+1}}{\binom{m}{n}} = \frac{\Gamma(m+1)\Gamma(n+1)\Gamma(m-n+1)}{\Gamma(n+2)\Gamma(m-n)\Gamma(m)} = \frac{-n+m}{n+1}$$

- Given a hypergeometric term $g$, is there hypergeometric term $f$ such that $\Delta f = g$?

$$f_{n+1} - f_n = g_n$$

# *Gospers algorithm*

- Gospers algorithm for indefinite hypergeometric summation determines $f_n$ from a given recursion

$$f_n = f_{n-1} + g_{n-1} = f_{n-2} + g_{n-1} + g_{n-2} = \cdots = f_0 + \sum_{k=0}^{n-1} g_k$$

  - Idea: recursive algorithm; telescoping

# *Gospers algorithm*

- Gospers algorithm for indefinite hypergeometric summation determines $f_n$ from a given recursion

$$f_n = f_{n-1} + g_{n-1} = f_{n-2} + g_{n-1} + g_{n-2} = \cdots = f_0 + \sum_{k=0}^{n-1} g_k$$

  - Idea: recursive algorithm; telescoping

- Ratio $\frac{f_n}{g_n} = \frac{f_n}{f_{n+1} - f_n} = \frac{1}{\frac{f_{n+1}}{f_n} - 1}$ is rational function of $n$

# *Gospers algorithm*

- Gospers algorithm for indefinite hypergeometric summation determines $f_n$ from a given recursion

$$f_n = f_{n-1} + g_{n-1} = f_{n-2} + g_{n-1} + g_{n-2} = \cdots = f_0 + \sum_{k=0}^{n-1} g_k$$

  - Idea: recursive algorithm; telescoping

- Ratio $\frac{f_n}{g_n} = \frac{f_n}{f_{n+1} - f_n} = \frac{1}{\frac{f_{n+1}}{f_n} - 1}$ is rational function of $n$

- Solve recursion with ansatz $f_n = y(n) g_n$ and (unknown) rational function $y(n)$

$$f_{n+1} - f_n = g_n \qquad \longrightarrow \qquad r(n)y(n+1) - y(n) = 1$$

# *Gospers algorithm*

- Gospers algorithm for indefinite hypergeometric summation determines $f_n$ from a given recursion

$$f_n = f_{n-1} + g_{n-1} = f_{n-2} + g_{n-1} + g_{n-2} = \cdots = f_0 + \sum_{k=0}^{n-1} g_k$$

  - Idea: recursive algorithm; telescoping

- Ratio $\frac{f_n}{g_n} = \frac{f_n}{f_{n+1} - f_n} = \frac{1}{\frac{f_{n+1}}{f_n} - 1}$ is rational function of $n$

- Solve recursion with ansatz $f_n = y(n) g_n$ and (unknown) rational function $y(n)$

$$f_{n+1} - f_n = g_n \qquad \longrightarrow \qquad r(n)y(n+1) - y(n) = 1$$

- *Upshot*
  - Solve *first-order linear recursion* for $y(n)$

# *Gospers algorithm (cont'd)*

- Given $f_{n+1} - f_n = g_n$ and ansatz $f_n = y(n)g_n$ with rational function $y(n)$, then $r(n)y(n+1) - y(n) = 1$

# *Gospers algorithm (cont'd)*

- Given $f_{n+1} - f_n = g_n$ and ansatz $f_n = y(n)g_n$ with rational function $y(n)$, then $r(n)y(n+1) - y(n) = 1$
- Let $r(n) = \frac{a(n)}{b(n)} \frac{c(n+1)}{c(n)}$ with polynomials $a(n), b(n), c(n)$ and $\gcd(a(n), b(n+k)) = 1$

# *Gospers algorithm (cont'd)*

- Given $f_{n+1} - f_n = g_n$ and ansatz $f_n = y(n)g_n$ with rational function $y(n)$, then $r(n)y(n+1) - y(n) = 1$
- Let $r(n) = \frac{a(n)}{b(n)} \frac{c(n+1)}{c(n)}$ with polynomials $a(n), b(n), c(n)$ and $\gcd(a(n), b(n+k)) = 1$
- Ansatz for $y(n)$ becomes $y(n) = \frac{b(n-1)}{c(n)} x(n)$ with (unknown) polynomial $x(n)$

# *Gospers algorithm (cont'd)*

- Given $f_{n+1} - f_n = g_n$ and ansatz $f_n = y(n)g_n$ with rational function $y(n)$, then $r(n)y(n+1) - y(n) = 1$
- Let $r(n) = \frac{a(n)}{b(n)} \frac{c(n+1)}{c(n)}$ with polynomials $a(n), b(n), c(n)$ and $\gcd(a(n), b(n+k)) = 1$
- Ansatz for $y(n)$ becomes $y(n) = \frac{b(n-1)}{c(n)} x(n)$ with (unknown) polynomial $x(n)$
- Solve for non-zero $x(n)$

$$a(n)x(n+1) - b(n-1)x(n) = c(n)$$

If non-zero $x(n)$ exists, hypergeometric recursion is summable.

# *(In)definite summation*

- Note the difference between indefinite and definite summation and integration

## *Indefinite summation*

- Sum and summand have the same number of variables

$$g(n,m) \quad = \quad \sum_{k=0}^{n} f(k,m)$$

## *Definite summation*

- Sum has one variable less than the summand

$$g(m) \quad = \quad \sum_{k=0}^{\infty} f(k,m)$$

- Situation for integration is fully analogous

# *Wilf-Zeilberger algorithm*

- WZ algorithm
  - definite hypergeometric summation
  - telescoping

# *Wilf-Zeilberger algorithm*

- WZ algorithm
  - definite hypergeometric summation
  - telescoping

# *Examples*

- Definite vs. indefinite summation

$$\sum_k \binom{n}{k} = \sum_k \binom{n}{k}$$

$$\sum_{k=0}^{n} \binom{n}{k} = 2^n$$

# *Harmonic summation*

- Harmonic sums $S_{m_1,\ldots,m_k}(n)$
  Gonzalez-Arroyo, Lopez, Ynduráin '79; Vermaseren '98; S.M., Uwer, Weinzierl '01
  - recursive definition $S_{\pm m_1,\ldots,m_k}(n) = \sum_{i=1}^{n} \frac{(\pm 1)^i}{i^{m_1}} S_{m_2,\ldots,m_k}(i)$

# *Harmonic summation*

- Harmonic sums $S_{m_1,\ldots,m_k}(n)$
  Gonzalez-Arroyo, Lopez, Ynduráin '79; Vermaseren '98; S.M., Uwer, Weinzierl '01
  - recursive definition $S_{\pm m_1,\ldots,m_k}(n) = \sum_{i=1}^{n} \frac{(\pm 1)^i}{i^{m_1}} S_{m_2,\ldots,m_k}(i)$
- Expansion of Gamma-function in $\epsilon = 2 - \frac{D}{2}$ around positive integers values ($n \geq 0$)

$$\frac{\Gamma(n+1+\epsilon)}{\Gamma(1+\epsilon)} = \Gamma(n+1)\,\exp\left(-\sum_{k=1}^{\infty} \epsilon^k \frac{(-1)^k}{k} S_k(n)\right)$$

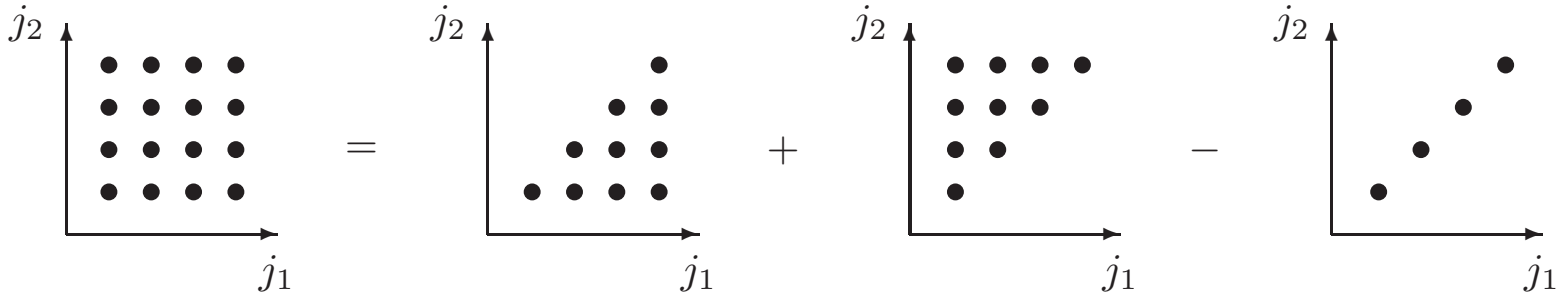
# *Algorithms for harmonic sums*

- Multiplication (Hopf algebra)
  - basic formula (recursion)

$$
\begin{aligned}
S_{m_1,\ldots,m_k}(n) \times S_{m'_1,\ldots,m'_l}(n) \quad = \quad & \sum_{j_1=1}^{n} \frac{1}{j_1^{m_1}} S_{m_2,\ldots,m_k}(j_1) S_{m'_1,\ldots,m'_l}(j_1) \\
& + \sum_{j_2=1}^{n} \frac{1}{j_2^{m'_1}} S_{m_1,\ldots,m_k}(j_2) S_{m'_2,\ldots,m'_l}(j_2) \\
& - \sum_{j=1}^{n} \frac{1}{j^{m_1+m'_1}} S_{m_2,\ldots,m_k}(j) S_{m'_2,\ldots,m'_l}(j)
\end{aligned}
$$

- Proof uses decomposition

$$
\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} \quad = \quad \sum_{i=1}^{n}\sum_{j=1}^{i} a_{ij} + \sum_{j=1}^{n}\sum_{i=1}^{j} a_{ij} - \sum_{i=1}^{n} a_{ii}
$$



# *Algorithms for harmonic sums (cont'd)*

- Convolution (sum over $n-j$ and $j$)

$$\sum_{j=1}^{n-1} \frac{1}{j^{m_1}} S_{m_2,\ldots,m_k}(j) \frac{1}{(n-j)^{n_1}} S_{n_2,\ldots,n_l}(n-j)$$

- Conjugation

$$-\sum_{j=1}^{n} \binom{n}{j} (-1)^j \frac{1}{j^{m_1}} S_{m_2,\ldots,m_k}(j)$$

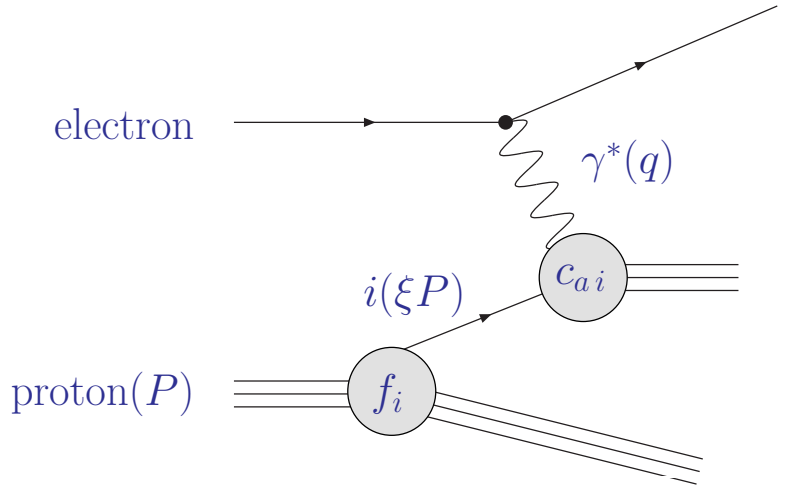
- Binomial convolution (sum over binomial, $n-j$ and $j$)

$$-\sum_{j=1}^{n-1} \binom{n}{j} (-1)^j \frac{1}{j^{m_1}} S_{m_2,\ldots,m_k}(j) \frac{1}{(n-j)^{n_1}} S_{n_2,\ldots,n_l}(n-j)$$

# *Deep-inelastic scattering*

## *QCD corrections to structure functions*

- Cross section $\sigma \simeq L^{\mu\nu} W_{\mu\nu}$ with leptonic tensor $L_{\mu\nu}$
  - hadronic tensor $L_{\mu\nu}$ parametrized through structure functions




- Structure function $F_2$ (up to terms $\mathcal{O}(1/Q^2)$ )
  - coefficient functions $c_a$ (hard scattering)

$$F_2(x, Q^2) \;=\; x \sum_i \int_x^1 \frac{d\xi}{\xi}\, c_{2,i}\left(\frac{x}{\xi}, \alpha_s(\mu^2), \frac{\mu^2}{Q^2}\right) f_i(\xi, \mu^2)$$

# *Mellin moments*

- Cross section $\sigma(N)$ in Mellin $N$-space

$$\begin{aligned}
\sigma(N) &= \int_0^1 dx\, x^N \frac{d\sigma(x)}{dx} \\
&= \int_0^1 dx\, x^N \int dLIPS^{(m)} \left|\mathcal{A}(\{\mathrm{in}\} \to \{\mathrm{out}\})\right|^2 \delta(x - f(p)) \\
&= \int dLIPS^{(m)} \left|\mathcal{A}(\{\mathrm{in}\} \to \{\mathrm{out}\})\right|^2 \frac{1}{(f)^{-N}}
\end{aligned}$$

  - $f(p)$ an invariant depended on internal and external momenta

- Upshot
  - mapping to discrete set of variables (positive integer Mellin $N$)

# *Reductions in Mellin space*

- Reductions with integration-by-parts in Mellin space
- $N$ case leads to monomials $\prod_i \frac{1}{P_i(\epsilon)^{\nu_i - N}}$
  - $P_i$ relevant propagators in the problem
  - $\nu_i$ fixed integers

## *Reductions in Mellin space*

- Reductions with integration-by-parts in Mellin space
- $N$ case leads to monomials $\prod_i \frac{1}{P_i(\epsilon)^{\nu_i - N}}$
  - $P_i$ relevant propagators in the problem
  - $\nu_i$ fixed integers

## *Integration-by-parts*

- Integration-by-parts exploits translational invariance
  - in $p$-space

$$0 = \int d^D p \, \frac{\partial}{\partial p^\mu} f(p, \dots)$$

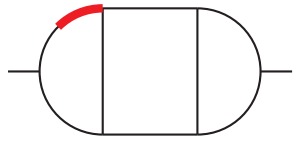
  - in $x$-space

$$0 = (x_1 - x_2)_\mu + (x_2 - x_3)_\mu + (x_3 - x_1)_\mu$$

# *Example from DIS*

- Scalar diagram with external momenta $P$ and $Q$ (four-point function with underlying ladder topology)

  coefficient $\frac{(2P \cdot p_1)^N}{(p_1)^N}$

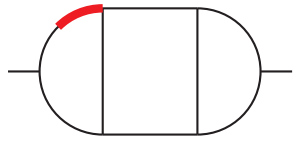


$$= \int \prod_n^3 d^D p_n \frac{1}{(P-p_1)^2} \frac{1}{p_1^2 \ldots p_8^2}$$

# *Example from DIS*

- Scalar diagram with external momenta $P$ and $Q$ (four-point function with underlying ladder topology)

  coefficient $\dfrac{(2P\cdot p_1)^N}{(p_1)^N}$



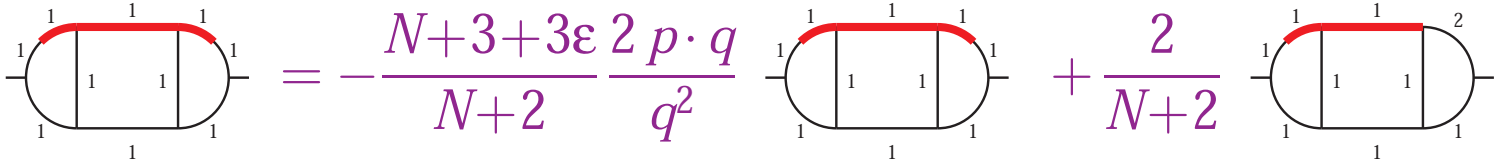
$$= \int \prod_n^3 d^D p_n \, \frac{1}{(P-p_1)^2} \, \frac{1}{p_1^2 \ldots p_8^2}$$

- $N$-th moment:
  coefficient of $(2P\cdot Q)^N$

$$= \frac{(2\,P\cdot Q)^N}{(Q^2)^{N+\alpha}} \, C_N$$

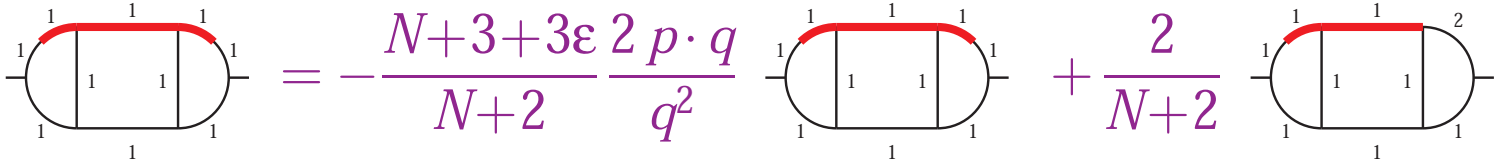
# *Difference equations*

- Single-step difference equation in $N$
  - extremely simple example



# *Difference equations*

- Single-step difference equation in $N$
  - extremely simple example

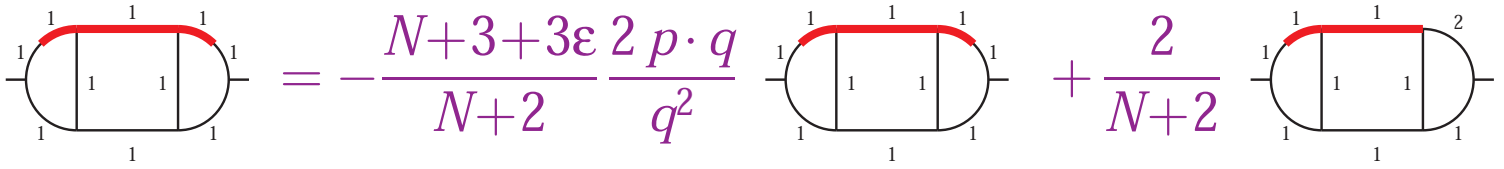



- Formal equation

$$\mathbf{I(N)} \quad = \quad -\frac{N+3+3\epsilon}{N+2}\,\mathbf{I(N-1)} + \frac{2}{N+2}\,\mathbf{G(N)}$$

# *Difference equations*

- Single-step difference equation in $N$
  - extremely simple example

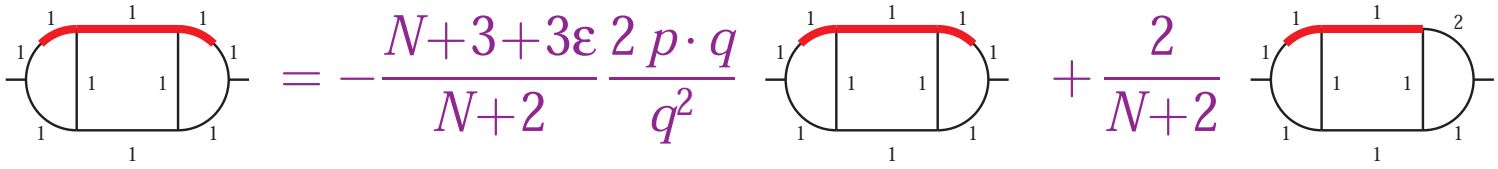



- Formal equation, formal solution

$$\mathbf{I(N)} = (-1)^N \frac{\prod_{j=1}^{N}(j+3+3\epsilon)}{\prod_{j=1}^{N}(j+2)}\mathbf{I(0)} + (-1)^N \sum_{\mathbf{i=1}}^{\mathbf{N}} (-1)^j \frac{\prod_{j=i+1}^{N}(j+3+3\epsilon)}{\prod_{j=i}^{N}(j+2)}\mathbf{G(i)}$$

# *Difference equations*

- Single-step difference equation in $N$
  - extremely simple example




- Formal equation, formal solution, input to solution

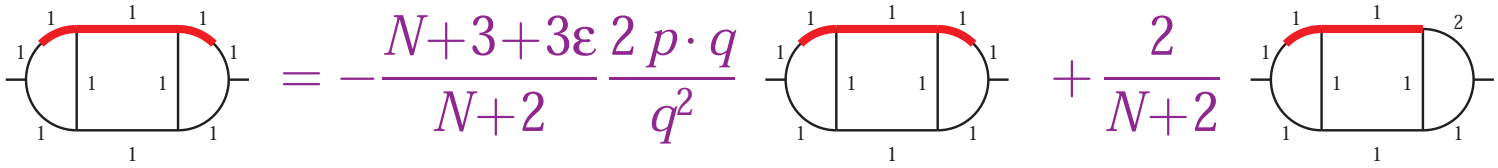
$$\mathbf{I(N)} = (-1)^N \frac{\prod_{j=1}^{N}(j+3+3\epsilon)}{\prod_{j=1}^{N}(j+2)}\mathbf{I(0)} + (-1)^N \sum_{\mathbf{i=1}}^{\mathbf{N}} (-1)^j \frac{\prod_{j=i+1}^{N}(j+3+3\epsilon)}{\prod_{j=i}^{N}(j+2)}\mathbf{G(i)}$$

$$\mathbf{I(0)} = -\frac{2}{3}\frac{1}{\epsilon^2} + \frac{23}{3}\frac{1}{\epsilon} - 42$$

$$\mathbf{G(i)} = \frac{(-1)^i}{\epsilon^2}\frac{2}{3}\left(\frac{S_1(i+2)}{i+2} - \frac{S_{1,2}(i)}{2} - \frac{S_2(i+1)}{2(i+1)} - S_2(i) - \frac{1}{(i+1)^2} - \frac{1}{(i+2)^2}\right) + \ldots$$

# *Difference equations*

- Single-step difference equation in $N$
  - extremely simple example




- Formal equation, formal solution, input to solution

$$\mathbf{I(N)} = (-1)^N \frac{\prod_{j=1}^{N}(j+3+3\epsilon)}{\prod_{j=1}^{N}(j+2)}\mathbf{I(0)} + (-1)^N \sum_{\mathbf{i=1}}^{\mathbf{N}} (-1)^j \frac{\prod_{j=i+1}^{N}(j+3+3\epsilon)}{\prod_{j=i}^{N}(j+2)}\mathbf{G(i)}$$

$$\mathbf{I(0)} = -\frac{2}{3}\frac{1}{\epsilon^2} + \frac{23}{3}\frac{1}{\epsilon} - 42$$

$$\mathbf{G(i)} = \frac{(-1)^i}{\epsilon^2}\frac{2}{3}\left(\frac{S_1(i+2)}{i+2} - \frac{S_{1,2}(i)}{2} - \frac{S_2(i+1)}{2(i+1)} - S_2(i) - \frac{1}{(i+1)^2} - \frac{1}{(i+2)^2}\right) + \ldots$$

- *Upshot*
  - automatic build-up of *nested sums*
  - efficient implementation in FORM

**I(N) =**

```
sign(N)*ep -2 * (  /3*S(R(1),1 + N)*den(1 + N) +  /3*S(R(1),1 + N)*den(
  1 + N) 2 +  /3*S(R(1),2 + N)*den(2 + N) +  /3*S(R(1),2 + N)*den(2 + N) 2 +  /3*
  S(R(1),N) + 2/3*S(R(1,2),N) + 2/3*S(R(2),1 + N)*den(1 + N) + 2/3*S(R(2),2 + N)*
  den(2 + N) - 2*S(R(2),N) -  /3*S(R(2),N)*N +  *S(R(2,1),N) +  /3*S(R(2,1),N)*N
   -  *S(R(3),N) - 2*S(R(3),N)*N -  /3*den(1 + N) 2 -  *den(1 + N) 3 -  /3*den(2
   + N) 2 - 2*den(2 + N) 3 )

+ sign(N)*ep -1 * (  - 1 *S(R(1),1 + N)*den(1 + N) -   /3*S(R(1),1 + N)*
  den(1 + N) 2 - 2 /3*S(R(1),1 + N)*den(1 + N) 3 - 1 *S(R(1),2 + N)*den(2 + N) -
    /3*S(R(1),2 + N)*den(2 + N) 2 - 1 /3*S(R(1),2 + N)*den(2 + N) 3 - 2 *S(R(1),N)
   +  /3*S(R(1,1),1 + N)*den(1 + N) +  /3*S(R(1,1),1 + N)*den(1 + N) 2 +  /3*S(R(1
  ,1),2 + N)*den(2 + N) +  /3*S(R(1,1),N) + 1 /3*S(R(1,1,2),N) + 1 /3*S(R(1,2),1
   + N)*den(1 + N) + 1 /3*S(R(1,2),2 + N)*den(2 + N) - 1 *S(R(1,2),N) -  *S(R(1,2)
  ,N)*N + 1 *S(R(1,2,1),N) +  *S(R(1,2,1),N)*N - 2 *S(R(1,3),N) -  *S(R(1,3),N)*N
   -   /3*S(R(2),1 + N)*den(1 + N) -   /3*S(R(2),1 + N)*den(1 + N) 2 -   /3*S(R(2)
  ,2 + N)*den(2 + N) -  *S(R(2),2 + N)*den(2 + N) 2 +   /3*S(R(2),N) + 2 *S(R(2),N
  )*N + 1 *S(R(2,1),1 + N)*den(1 + N) +  *S(R(2,1),2 + N)*den(2 + N) - 13 /3*S(R(2
  ,1),N) -   /3*S(R(2,1),N)*N + 1 /3*S(R(2,1,1),N) +  /3*S(R(2,1,1),N)*N -  2/3*S(
  R(2,2),N) - 22/3*S(R(2,2),N)*N - 1 *S(R(3),1 + N)*den(1 + N) - 12*S(R(3),2 + N)*
  den(2 + N) +   *S(R(3),N) + 1  /3*S(R(3),N)*N - 1 *S(R(3,1),N) - 1 /3*S(R(3,1),N
  )*N + 3 *S(R( ),N) + 12*S(R( ),N)*N + 32*den(1 + N) 2 + 1  /3*den(1 + N) 3 + 2 *
  den(1 + N)   + 1 *den(2 + N) 2 +  2/3*den(2 + N) 3 + 12*den(2 + N)   )

+ sign(N) * ( 1  *S(R(1),1 + N)*den(1 + N) + 1  *S(R(1),1 + N)*den(1 + N)
   2 + 2  /3*S(R(1),1 + N)*den(1 + N) 3 - 1 /3*S(R(1),1 + N)*den(1 + N)   + 1  *S(
  R(1),2 + N)*den(2 + N) +   *S(R(1),2 + N)*den(2 + N) 2 + 13 /3*S(R(1),2 + N)*
  den(2 + N) 3 -  /3*S(R(1),2 + N)*den(2 + N)   + 1  *S(R(1),N) - 32*S(R(1,1),1 +
  N)*den(1 + N) -   /3*S(R(1,1),1 + N)*den(1 + N) 2 - 2 /3*S(R(1,1),1 + N)*den(1
   + N) 3 - 32*S(R(1,1),2 + N)*den(2 + N) -  /3*S(R(1,1),2 + N)*den(2 + N) 2 - 1 /
  3*S(R(1,1),2 + N)*den(2 + N) 3 -   *S(R(1,1),N) +  /3*S(R(1,1,1),1 + N)*den(1 +
  N) -   /3*S(R(1,1,1),1 + N)*den(1 + N) 2 +  /3*S(R(1,1,1),2 + N)*den(2 + N) -   /
  3*S(R(1,1,1),2 + N)*den(2 + N) 2 +  /3*S(R(1,1,1),N) + 3 /3*S(R(1,1,1,2),N) + 3 /
  3*S(R(1,1,2),1 + N)*den(1 + N) + 3 /3*S(R(1,1,2),2 + N)*den(2 + N) -   *S(R(1,1,
  2),N) - 12*S(R(1,1,2),N)*N +  2*S(R(1,1,2,1),N) + 12*S(R(1,1,2,1),N)*N -   *S(R(
  1,1,3),N) - 1 *S(R(1,1,3),N)*N - 1  /3*S(R(1,2),1 + N)*den(1 + N) +   /3*S(R(1,2
  ),1 + N)*den(1 + N) 2 - 13 /3*S(R(1,2),2 + N)*den(2 + N) + 1 *S(R(1,2),2 + N)*
  den(2 + N) 2 +  3 /3*S(R(1,2),N) +   *S(R(1,2),N)*N + 3 *S(R(1,2,1),1 + N)*den(1
   + N) + 1 *S(R(1,2,1),2 + N)*den(2 + N) -   2/3*S(R(1,2,1),N) -   *S(R(1,2,1),N)
  *N +   /3*S(R(1,2,1,1),N) +  *S(R(1,2,1,1),N)*N - 2  /3*S(R(1,2,2),N) - 22*S(R(1
  ,2,2),N)*N -   *S(R(1,3),1 + N)*den(1 + N) -   *S(R(1,3),2 + N)*den(2 + N) +    /
3*S(R(1,3),N) + 1  *S(R(1,3),N)*N - 11 /3*S(R(1,3,1),N) - 1 *S(R(1,3,1),N)*N +
 1 /3*S(R(1, ),N) + 3 *S(R(1, ),N)*N + 1  *S(R(2),1 + N)*den(1 + N) +    /3*S(R(
2),1 + N)*den(1 + N) 2 + 1  /3*S(R(2),1 + N)*den(1 + N) 3 + 13 *S(R(2),2 + N)*
den(2 + N) + 2  /3*S(R(2),2 + N)*den(2 + N) 2 +   /3*S(R(2),2 + N)*den(2 + N) 3
 -   *S(R(2),N) - 1  *S(R(2),N)*N - 33 /3*S(R(2,1),1 + N)*den(1 + N) -   /3*S(R(
2,1),1 + N)*den(1 + N) 2 - 2  /3*S(R(2,1),2 + N)*den(2 + N) - 1 /3*S(R(2,1),2 +
N)*den(2 + N) 2 +    /3*S(R(2,1),N) + 1  *S(R(2,1),N)*N +   /3*S(R(2,1,1),1 + N)
*den(1 + N) + 2 /3*S(R(2,1,1),2 + N)*den(2 + N) - 1  /3*S(R(2,1,1),N) - 1  /3*S(
R(2,1,1),N)*N - 12*S(R(2,1,1,1),N) +  /3*S(R(2,1,1,1),N)*N + 3 /3*S(R(2,1,2),N)
 - 2/3*S(R(2,1,2),N)*N - 1 2/3*S(R(2,2),1 + N)*den(1 + N) - 11 /3*S(R(2,2),2 + N
)*den(2 + N) +    /3*S(R(2,2),N) + 3  /3*S(R(2,2),N)*N - 11 /3*S(R(2,2,1),N) -
1 *S(R(2,2,1),N)*N + 2  /3*S(R(2,3),N) + 3 *S(R(2,3),N)*N +    /3*S(R(3),1 + N)*
den(1 + N) + 1  /3*S(R(3),1 + N)*den(1 + N) 2 +    /3*S(R(3),2 + N)*den(2 + N)
 +   /3*S(R(3),2 + N)*den(2 + N) 2 - 1   /3*S(R(3),N) - 2  *S(R(3),N)*N -   /3*
S(R(3,1),1 + N)*den(1 + N) -   /3*S(R(3,1),2 + N)*den(2 + N) +    /3*S(R(3,1),N)
 + 1  /3*S(R(3,1),N)*N - 1 *S(R(3,1,1),N) - 1 /3*S(R(3,1,1),N)*N +   *S(R(3,2),N
) +   /3*S(R(3,2),N)*N + 3 2/3*S(R( ),1 + N)*den(1 + N) + 1  /3*S(R( ),2 + N)*
den(2 + N) -  3 *S(R( ),N) -    /3*S(R( ),N)*N -  *S(R( ,1),N) -  /3*S(R( ,1),N)
*N - 1  *S(R( ),N) -   *S(R( ),N)*N - 2  *den(1 + N) 2 - 3  *den(1 + N) 3 -    /
3*den(1 + N)   - 1  *den(1 + N)   - 1  *den(2 + N) 2 - 1  *den(2 + N) 3 -    /3*
den(2 + N)   -   *den(2 + N)   );
```

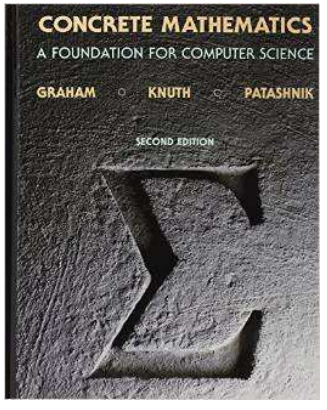
**I(N) =**

```
sign(N)*ep -2 * (  /3*S(R(1),1 + N)*den(1 + N) +  /3*S(R(1),1 + N)*den(
  1 + N) 2 +  /3*S(R(1),2 + N)*den(2 + N) +  /3*S(R(1),2 + N)*den(2 + N) 2 +  /3*
  S(R(1),N) + 2/3*S(R(1,2),N) + 2/3*S(R(2),1 + N)*den(1 + N) + 2/3*S(R(2),2 + N)*
  den(2 + N) - 2*S(R(2),N) -  /3*S(R(2),N)*N +  *S(R(2,1),N) +  /3*S(R(2,1),N)*N
   -  *S(R(3),N) - 2*S(R(3),N)*N -  /3*den(1 + N) 2 -  *den(1 + N) 3 -  /3*den(2
   + N) 2 - 2*den(2 + N) 3 )

+ sign(N)*ep -1 * (  - 1 *S(R(1),1 + N)*den(1 + N) -   /3*S(R(1),1 + N)*
  den(1 + N) 2 - 2 /3*S(R(1),1 + N)*den(1 + N) 3 - 1 *S(R(1),2 + N)*den(2 + N) -
    /3*S(R(1),2 + N)*den(2 + N) 2 - 1 /3*S(R(1),2 + N)*den(2 + N) 3 - 2 *S(R(1),N)
   +  /3*S(R(1,1),1 + N)*den(1 + N) +  /3*S(R(1,1),1 + N)*den(1 + N) 2 +  /3*S(R(1
  ,1),2 + N)*den(2 + N) +  /3*S(R(1,1),N) + 1 /3*S(R(1,1,2),N) + 1 /3*S(R(1,2),1
   + N)*den(1 + N) + 1 /3*S(R(1,2),2 + N)*den(2 + N) - 1 *S(R(1,2),N) -  *S(R(1,2)
  ,N)*N + 1 *S(R(1,2,1),N) +  *S(R(1,2,1),N)*N - 2 *S(R(1,3),N) -  *S(R(1,3),N)*N
   -   /3*S(R(2),1 + N)*den(1 + N) -   /3*S(R(2),1 + N)*den(1 + N) 2 -   /3*S(R(2)
  ,2 + N)*den(2 + N) -  *S(R(2),2 + N)*den(2 + N) 2 +   /3*S(R(2),N) + 2 *S(R(2),N
  )*N + 1 *S(R(2,1),1 + N)*den(1 + N) +  *S(R(2,1),2 + N)*den(2 + N) - 13 /3*S(R(2
  ,1),N) -   /3*S(R(2,1),N)*N + 1 /3*S(R(2,1,1),N) +  /3*S(R(2,1,1),N)*N -  2/3*S(
  R(2,2),N) - 22/3*S(R(2,2),N)*N - 1 *S(R(3),1 + N)*den(1 + N) - 12*S(R(3),2 + N)*
  den(2 + N) +   *S(R(3),N) + 1  /3*S(R(3),N)*N - 1 *S(R(3,1),N) - 1 /3*S(R(3,1),N
  )*N + 3 *S(R( ),N) + 12*S(R( ),N)*N + 32*den(1 + N) 2 + 1  /3*den(1 + N) 3 + 2 *
  den(1 + N)   + 1 *den(2 + N) 2 +  2/3*den(2 + N) 3 + 12*den(2 + N)   )

+ sign(N) * ( 1  *S(R(1),1 + N)*den(1 + N) + 1  *S(R(1),1 + N)*den(1 + N)
   2 + 2  /3*S(R(1),1 + N)*den(1 + N) 3 - 1 /3*S(R(1),1 + N)*den(1 + N)   + 1  *S(
  R(1),2 + N)*den(2 + N) +   *S(R(1),2 + N)*den(2 + N) 2 + 13 /3*S(R(1),2 + N)*
  den(2 + N) 3 -  /3*S(R(1),2 + N)*den(2 + N)   + 1  *S(R(1),N) - 32*S(R(1,1),1 +
  N)*den(1 + N) -   /3*S(R(1,1),1 + N)*den(1 + N) 2 - 2 /3*S(R(1,1),1 + N)*den(1
   + N) 3 - 32*S(R(1,1),2 + N)*den(2 + N) -  /3*S(R(1,1),2 + N)*den(2 + N) 2 - 1 /
  3*S(R(1,1),2 + N)*den(2 + N) 3 -   *S(R(1,1),N) +  /3*S(R(1,1,1),1 + N)*den(1 +
  N) -   /3*S(R(1,1,1),1 + N)*den(1 + N) 2 +  /3*S(R(1,1,1),2 + N)*den(2 + N) -   /
  3*S(R(1,1,1),2 + N)*den(2 + N) 2 +  /3*S(R(1,1,1),N) + 3 /3*S(R(1,1,1,2),N) + 3 /
  3*S(R(1,1,2),1 + N)*den(1 + N) + 3 /3*S(R(1,1,2),2 + N)*den(2 + N) -   *S(R(1,1,
  2),N) - 12*S(R(1,1,2),N)*N +  2*S(R(1,1,2,1),N) + 12*S(R(1,1,2,1),N)*N -   *S(R(
  1,1,3),N) - 1 *S(R(1,1,3),N)*N - 1  /3*S(R(1,2),1 + N)*den(1 + N) +   /3*S(R(1,2
  ),1 + N)*den(1 + N) 2 - 13 /3*S(R(1,2),2 + N)*den(2 + N) + 1 *S(R(1,2),2 + N)*
  den(2 + N) 2 +  3 /3*S(R(1,2),N) +   *S(R(1,2),N)*N + 3 *S(R(1,2,1),1 + N)*den(1
   + N) + 1 *S(R(1,2,1),2 + N)*den(2 + N) -   2/3*S(R(1,2,1),N) -   *S(R(1,2,1),N)
  *N +   /3*S(R(1,2,1,1),N) +  *S(R(1,2,1,1),N)*N - 2  /3*S(R(1,2,2),N) - 22*S(R(1
  ,2,2),N)*N -   *S(R(1,3),1 + N)*den(1 + N) -   *S(R(1,3),2 + N)*den(2 + N) +    /
3*S(R(1,3),N) + 1  *S(R(1,3),N)*N - 11 /3*S(R(1,3,1),N) - 1 *S(R(1,3,1),N)*N +
 1 /3*S(R(1, ),N) + 3 *S(R(1, ),N)*N + 1  *S(R(2),1 + N)*den(1 + N) +    /3*S(R(
2),1 + N)*den(1 + N) 2 + 1  /3*S(R(2),1 + N)*den(1 + N) 3 + 13 *S(R(2),2 + N)*
den(2 + N) + 2  /3*S(R(2),2 + N)*den(2 + N) 2 +   /3*S(R(2),2 + N)*den(2 + N) 3
 -   *S(R(2),N) - 1  *S(R(2),N)*N - 33 /3*S(R(2,1),1 + N)*den(1 + N) -   /3*S(R(
2,1),1 + N)*den(1 + N) 2 - 2  /3*S(R(2,1),2 + N)*den(2 + N) - 1 /3*S(R(2,1),2 +
N)*den(2 + N) 2 +    /3*S(R(2,1),N) + 1  *S(R(2,1),N)*N +   /3*S(R(2,1,1),1 + N)
*den(1 + N) + 2 /3*S(R(2,1,1),2 + N)*den(2 + N) - 1  /3*S(R(2,1,1),N) - 1  /3*S(
R(2,1,1),N)*N - 12*S(R(2,1,1,1),N) +  /3*S(R(2,1,1,1),N)*N + 3 /3*S(R(2,1,2),N)
 - 2/3*S(R(2,1,2),N)*N - 1 2/3*S(R(2,2),1 + N)*den(1 + N) - 11 /3*S(R(2,2),2 + N
)*den(2 + N) +    /3*S(R(2,2),N) + 3  /3*S(R(2,2),N)*N - 11 /3*S(R(2,2,1),N) -
1 *S(R(2,2,1),N)*N + 2  /3*S(R(2,3),N) + 3 *S(R(2,3),N)*N +    /3*S(R(3),1 + N)*
den(1 + N) + 1  /3*S(R(3),1 + N)*den(1 + N) 2 +    /3*S(R(3),2 + N)*den(2 + N)
 +   /3*S(R(3),2 + N)*den(2 + N) 2 - 1   /3*S(R(3),N) - 2  *S(R(3),N)*N -   /3*
S(R(3,1),1 + N)*den(1 + N) -   /3*S(R(3,1),2 + N)*den(2 + N) +    /3*S(R(3,1),N)
 + 1  /3*S(R(3,1),N)*N - 1 *S(R(3,1,1),N) - 1 /3*S(R(3,1,1),N)*N +   *S(R(3,2),N
) +   /3*S(R(3,2),N)*N + 3 2/3*S(R( ),1 + N)*den(1 + N) + 1  /3*S(R( ),2 + N)*
den(2 + N) -  3 *S(R( ),N) -    /3*S(R( ),N)*N -  *S(R( ,1),N) -  /3*S(R( ,1),N)
*N - 1  *S(R( ),N) -   *S(R( ),N)*N - 2  *den(1 + N) 2 - 3  *den(1 + N) 3 -    /
3*den(1 + N)   - 1  *den(1 + N)   - 1  *den(2 + N) 2 - 1  *den(2 + N) 3 -    /3*
den(2 + N)   -   *den(2 + N)   );
```

- Result for $\mathbf{I(N)}$ in the G-scheme

# Literature (I)

- Text books
  - *Modern Computer Algebra*
    J. von zur Gathen, J. Gerhard
  - *Concrete Mathematics*
    R. L Graham, D. E. Knuth, O. Pataschnik
  - *Concrete Tetrahydron*
    M. Kauers, P. Paule
  - *A=B*
    M. Petkovsek, H. S. Wilf, D. Zeilberger `www.math.upenn.edu/~wilf/AeqB.html`



# *Literature (II)*

- Selected research articles
  - *Harmonic sums, Mellin transforms and integrals*, J. Vermaseren; hep-ph/9806280
  - *Nested sums, expansion of transcendental functions and multi-scale multi-loop integrals*, S.M., P. Uwer, S. Weinzierl; hep-ph/0110083
  - *Gauss hypergeometric function: Reduction, epsilon-expansion for integer/half-integer parameters and Feynman diagrams*, M. Yu. Kalmykov; hep-th/0602028
  - *HypExp 2, Expanding Hypergeometric Functions about Half-Integer Parameters*, T. Huber, D. Maitre; 0708.2443
  - *On the analytic computation of massless propagators in dimensional regularization*, E. Panzer; 1305.2161
  - ...

# *Software (I)*

## *Requirements in particle physics*

- Symbolic calculations characterized by need for basic operations
  - sorting, gcd, factorization, multiplication
  - symbolic integration/summation
  - solution of systems of equations
  - ...
- Specialized code usually written by the user
  - largely dependent on the physics problem
  - add-on libraries
- No need for a system which knows more than the user

# *Software (II)*

- Commercial programs: *Mathematica*, *Maple*, . . .
- Freeware/Add-on packages
  - *Mathematica*, *Maple*
    - several packages for hypergeometric summation
      [see for instance `www.math.upenn.edu/~wilf/AeqB.html`]
    - RISC software for symbolic summation and integration
      `www.risc.jku.at/research/combinat/software`
    - expansion of hypergeometric functions *HypExp*, T. Huber, D. Maitre
    - reduction of hypergeometric functions *HyperDire*, V. Bytev
    - hyperlogarithmic integration *hyperint*, E. Panzer
  - GiNaC `www.ginac.de`
    - *nestedsums*, S. Weinzierl
  - Form `www.nikhef.nl/~form`
    - *Summer6*, J. Vermaseren
    - *XSummer*, S. Moch, P. Uwer

# *Exercises 1*

## *Polynomial summmation*

- Use *Mathematica* or *Maple* for polynomial summmation.
- Check some of the examples for hypergeometric summation with *Mathematica* or *Maple* like

$$\sum_{i\geq 0} a^{\overline{i}}\,\frac{z^i}{i!} \quad = \quad \frac{1}{(1-z)^a}$$

$$-z\sum_{i\geq 0} \frac{1^{\overline{i}}1^{\overline{i}}}{2^{\overline{i}}}\,\frac{z^i}{i!} \quad = \quad \ln(1-z)$$

- Try to evaluate the sum $\displaystyle\sum_{j_1=1}^{N} \frac{1}{j_1} S_1(j_1)$ in *Mathematica* or *Maple*.

  What happens?

# *Exercises 2*

## *Harmonic sums*

- Use the FORM package SUMMER for manipulation of harmonic sums. Download the package `summer.h` from
  http://www.nikhef.nl/~form/maindir/packages/summer/.
  - Evaluate the product of harmonic sums $S_2(N)S_1(N)^2$. Use the procedure `basis.prc`.
    ```
    #-
    #include summer.h
    .global
    L exampleproduct = S(R(2),N)*S(R(1),N)^2;
    #call basis(S)
    Print;
    .end
    ```
  - Check your result with the following sequence of calls.
    ```
    Multiply, replace_(N,<some_number>);
    #call subesses(S)
    ```

# *Exercises 3*

## *Harmonic summation (easy)*

- Use the FORM package SUMMER for harmonic summmation. Download the package `summer.h` from
  http://www.nikhef.nl/~form/maindir/packages/summer/.
  - Evaluate the sum $\sum_{j_1=1}^{N} \frac{1}{j_1} S_1(j_1)$ Use the procedure `summer.prc`.

```
#-
#include summer.h
.global
L examplesum = sum1(j1,1,N)*den(j1)*S(R(1),j1);
#call summer(1)
Print;
.end
```

  - Compute examples for the convolution and conjugation of harmonic sums. Use the notation `fac(N)`, `invfac(N)` and `sign(N)` with the (obvious) meaning $N!$, $\frac{1}{N!}$ and $(-1)^N$

# Exercises 4

## Harmonic summation (difficult)

- Use the FORM package XSUMMERfor summmation. Download the package from http://www-zeuthen.desy.de/~moch/xsummer/.
  - Evaluate the sum $\sum_{j_1=1}^{\infty} \frac{1}{(j_1+1)^3} \left(S_{-1}(j_1)\right)^2$

    Bailey, Borwein *Mathematics by Experiment*

```
#-
#define MAXSUM "2"
#define MAXWEIGHT "2"
#include declvars.h
#include declsums.h
L BBexample =
sum1(j1,1,inf)*(-S(R(1),X(-1),j1))^2*den(j1+1)^3;
#call DoSum(1,1)
id acc(x?)  = x;
Print +s;
.end
```

# *Exercises 5*

## *Bonus material*

- Use the FORM package `summer.h` to solve the difference equation

$$\mathbf{I(N)} \quad = \quad -\frac{N+3+3\epsilon}{N+2}\,\mathbf{I(N-1)} + \frac{2}{N+2}\,\mathbf{G(N)}$$

with the boundary conditions for $\mathbf{I(0)}$ and $\mathbf{G(N)}$ given in the lecture.